hakin9

PRACTICAL PROTECTION

IT SECURITY MAGAZINE

SECURE CODING

USING AMAZON AMI FOR CRACKING THE WPA2

STEGANO
– ENCRYP

SOCIAL E

VOIP HACKING

CACIES OF
DING IN JAVA

Vol.9 No.01
Issue 01/2014(70) ISSN: 1733-7186

PL

LENGE
: HOW TO SEND
EMAILS SECURELY

TEASER

Become a Big Data Master!

Over 45 HOW-TO, practical classes and tutorials to choose from!

Attend The 3rd Big Data TechCon!

The HOW-TO technical conference for professionals implementing Big Data



Come to Big Data TechCon to learn the best ways to:

- Process and analyze the real-time data pouring into your organization.
- Learn HOW TO integrate data collection technologies with data analytics and predictive analysis tools to produce the kind of workable information and reports your organization needs.
- Understand HOW TO leverage Big Data to help your organization today.
- Master Big Data tools and technologies like Hadoop, MapReduce, HBase, Cassandra, NoSQL databases, and more!
- Looking for Hadoop training? We have several Hadoop tutorials and dozens of Hadoop classes to get you started — or advanced classes to take you to the next level!

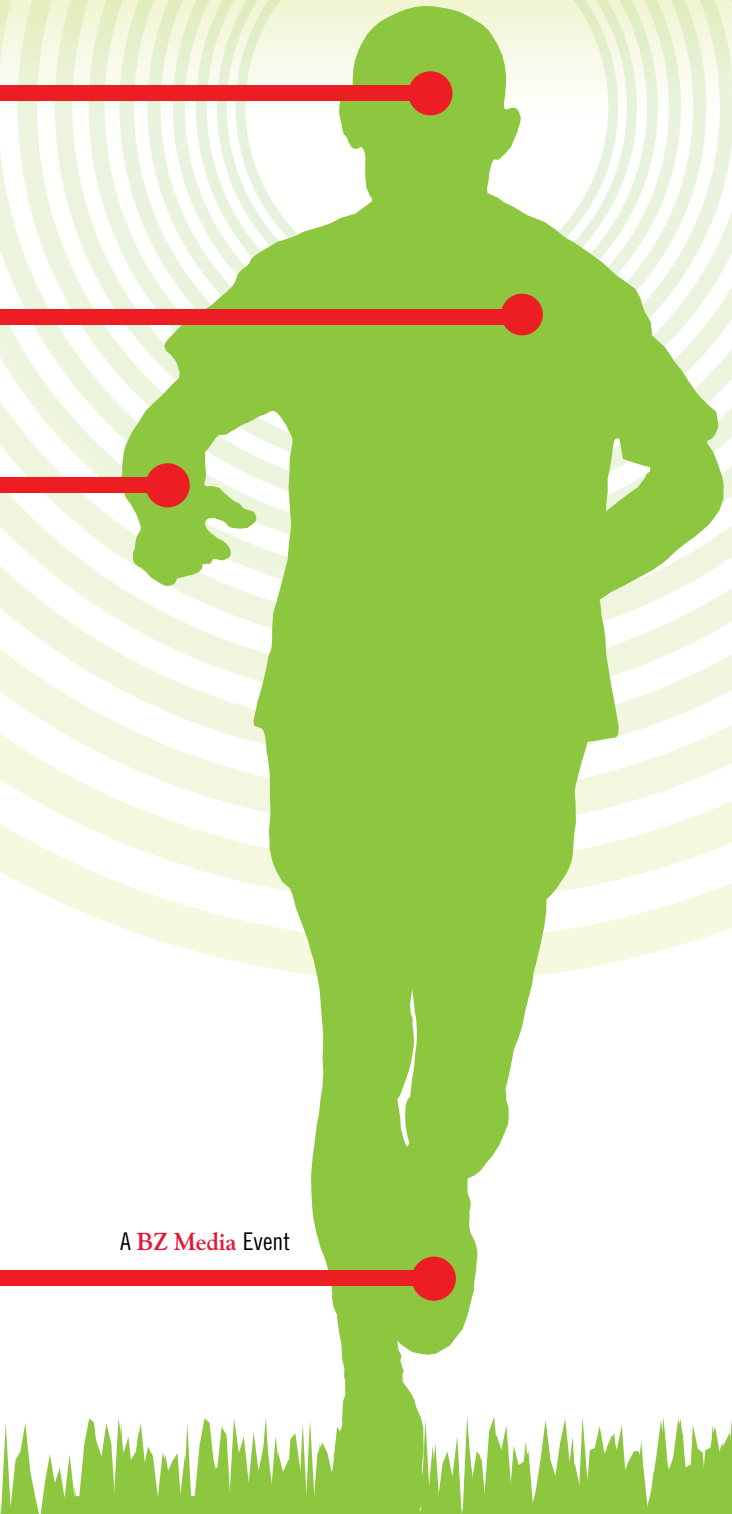
BigData TECHCON Boston

March 31-April 2, 2014

www.BigDataTechCon.com

A BZ Media Event Big Data TechCon

Join the Wearables Revolution!



Wearables DevCon

A conference for Designers, Builders and Developers of Wearable Computing Devices

Wearable computing devices are the Next Big Wave in technology. And the winning developers in the next decade are going to be the ones who take advantage of these new technologies EARLY and build the next generation of red-hot apps.

Choose from over 35 classes and tutorials!

- Learn how to develop apps for the coolest gadgets like Google Glass, FitBit, Pebble, the SmartWatch 2, Jawbone, and the Galaxy Gear SmartWatch
- Get practical answers to real problems, learn tangible steps to real-world implementation of the next generation of computing devices

March 5-7, 2014
San Francisco

WearablesDevCon.com

A BZ Media Event

Secure Coding

Table of Contents

Dear Hakin9 Readers!

We give you the new Hakin9 issue about practical protection. Less theory, more practice!

What will you find here?

We recommend you the article of Mr Mirko Raimondi about Voice over Internet Protocol hacking. VoIP is getting more and more popular way of voice communications, replacing the analog signal-working telephones. New technologies, new threats!

And have you ever been interested in steganoghaphy? Mr Sarang shows us how to encode the files they would look like another file than they are in reality. This is not just a trick – it is better to know that the hackers can send a hidden Trojan in file looking like image, video, message...

We have also a pleasure to publish an article by Mr Thanglalson Gangte, who took a 2nd position in our Best IT Blog Challenge! Have you heard about Pretty Good Privacy? It's an asymmetric encryption algorithm. Mr Gangte shares his foundings about PGP which let us encrypting our emails.

This and much more in this Hakin9. Wish you a good reading!

Hakin9 Magazine Team

Editor in Chief: Ewa Duranc
ewa.duranc@hakin9.org

Managing Editor: Aleksandra Olszewska
aleksandra.olszewska@hakin9.org

Betatesters & Proofreaders: Elliott Bujan, Amit Chugh, Jonus Gerrits, Mbella Ekoume, Greg Hanis, Gilles Lami, Elia Pinto, Sagar Rahalkar, Robin Schroeder, John Webb

Special Thanks to the Beta testers and Proofreaders who helped us with this issue. Without their assistance there would not be a Hakin9 Magazine.

Publisher: Paweł Marciniak

CEO: Ewa Dudzic
ewa.dudzic.@hakin9.org

Product Manager: Ewa Duranc
ewa.duranc@hakin9.org

Production Director: Andrzej Kuca
andrzej.kuca@hakin9.org

Art. Director: Ireneusz Pogroszewski
ireneusz.pogroszewski@hakin9.org
DTP: Ireneusz Pogroszewski

Publisher: Hakin9 Media sp. z o.o. SK
02-676 Warszawa, ul. Postępu 17D
NIP 95123253396
www.hakin9.org/en

Windows Registry Forensics (WRF) with Volatility Framework Beginners Guide

by Kapil Soni

This is guide especially for beginners who wants to learn about Windows Registry Forensics. Throughout this book you will learn how to do forensics and investigate Windows Registry. Less Theory – More Practice.

Before starting discussion about tools which is used in windows registry forensics, I would like to inform that what we are going to do & what we want to do.

In this book we are going to analyse memory which will be in 2 way i.e. live memory forensics & dumped memory forensics. We'll use two tools in this book for experimental purpose. Memory forensics is becoming very essential & useful task in digital forensics as well as incidence response.

When system is infected & compromised by attacks or viruses, investigator need to perform analysis & forensic investigation of particular system.

In this book I am going to demonstrate forensics analysis by using dumped memory forensics.

Firstly I would like to tell that what actually memory is or what it contains?

Memory contains lot of important as well as confidential information about users & system. Why we need memory forensics because to tackle with memory malware, criminal cases, intrusion analysis etc.

All activities which are done by attacker like any malicious or harmful activity are analysed by memory forensics. All these activities are stored in format of logs or data. Sometimes such things are also encrypted we need to perform analysis on basis of nature of data.

Let's begin our demonstration with tools.

DumpIt

DumpIt is a free memory dumping tool for Windows by MoonSols that can dump all the memory in just one click. DumpIt is a very powerful and useful tool for dump memory on Windows platform. DumpIt is a fusion of two tools, Win32 and Win64 combined into one executable.

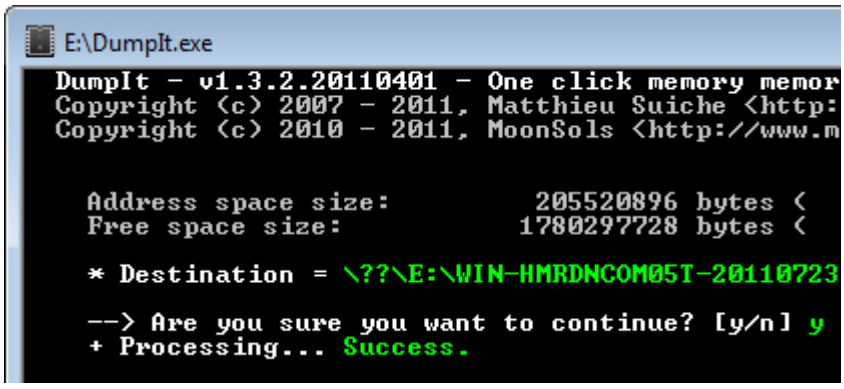


Figure 1. DumpIt Memory Dumper

Note

After all if you are having any problem in installation and usage, I recommend watch this video tutorials of How to use DumpIt: *http://www.youtube.com/watch?v=SEs4ZAolED0*.

Volatility Framework

Volatility framework which integrates almost digital forensics tools within it. The Volatility Framework is a completely open collection of tools, implemented in Python under the GNU General Public License, for the extraction of digital artifacts from volatile memory (RAM) samples. The extraction techniques are performed completely independent of the system being investigated but offer unprecedented visibility into the runtime state of the system. The framework is intended to introduce people to the techniques and complexities associated with extracting digital artifacts from volatile memory samples and provide a platform for further work into this exciting area of research.

Volatility supports memory dumps from all major 32- and 64-bit Windows versions and service packs including XP, 2003 Server, Vista, Server 2008, Server 2008 R2, and Seven. Whether your memory dump is in raw format, a Microsoft crash dump, hibernation file, or virtual machine snapshot, Volatility is able to work with it. We also now support Linux memory dumps in raw or LiME format and include 35+ plugins for analyzing 32- and 64- bit Linux kernels from 2.6.11 – 3.5.x and distributions such as Debian, Ubuntu, OpenSuSE, Fedora, CentOS, and Mandrake. We support 38 versions of Mac OSX memory dumps from 10.5 to 10.8.3 Mountain Lion, both 32- and 64-bit. Android phones with ARM processors are also supported. Support for Windows 8, 8.1, Server 2012, 2012 R2, and OSX 10.9 (Mavericks) is either already in svn or just around the corner, so stay tuned for our next release! Some advanced features of Volatility Framework is that provides tools for memory malware analysis, android memory analysis.

Chapter 2 – Basics of Memory Image (Dumped)

Dumped memory contains a lot of information which will be very useful as per forensic prospective. First of all we have to know about which memory dump we are going to analyse whether it is of Windows XP or Windows 7 because both are having different architecture. Some plugins of volatility will work on Windows XP not on Windows 7 & vice versa. First of all make sure, you have volatility framework standalone and you have a dumped memory file for forensics.

So let's start some practical things.

Image Information

In this section, we will see how to check basic information which contained in image (dumped file of windows XP). I'll use volatility framework to perform analysis of image. So I have "Volatility Framework Standalone Version" for Windows, and a dumped memory image of Windows XP.

How to get Image Information from image, Use this "imageinfo" plugin in your command prompt as follows:

```
Volatility.exe –f WinXP.raw imageinfo
```

Where,

Volatility.exe – is volatility framework that you can download from the internet.

-f – File Location/File Path

WinXP.raw – Dumped Windows memory image

Imageinfo – Volatility Framework plugin for check image information.

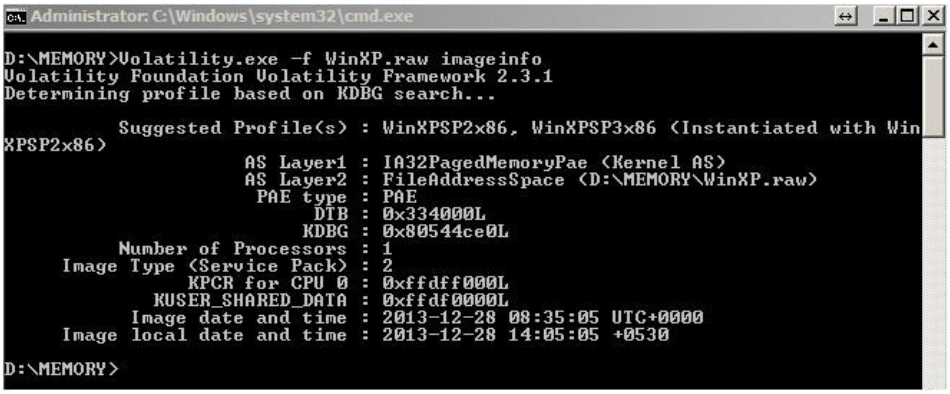


Figure 2. Image Information Gathering

After run “imageinfo” plugin you can see two profile suggested by it. Means it may be *WinXPSP2x86* or *WinXPSP3x86* profile containing this dumped image. I know my profile is WinXPSP2x86 so I’ll use it. You can figure out exact profile of dumped image by “kdbgscan” plugin, just use it instead of imageinfo and it will show you other interesting information.

Process Analysis

Process analysis is another most important part in memory forensics or in investigation because processes contains a lot of secrets and useful information as per forensic prospective. Take an example of memory malware analysis. Some malwares like Zeus, Lagma, Coreflood etc. To analyse such types of malwares, process analysis plays an important role. By analysing processes we can find out malwares which are there in memory. So again come back to our Windows registry forensics, we will discuss briefly about memory malware analysis in my next book.

To scan processes present in dumped memory image following plugin is used:

```
Volatility.exe -f WinXP.raw psscan
```

Where,

Psscan – Process scan plugin for print all the processes that was running on the system when image created.

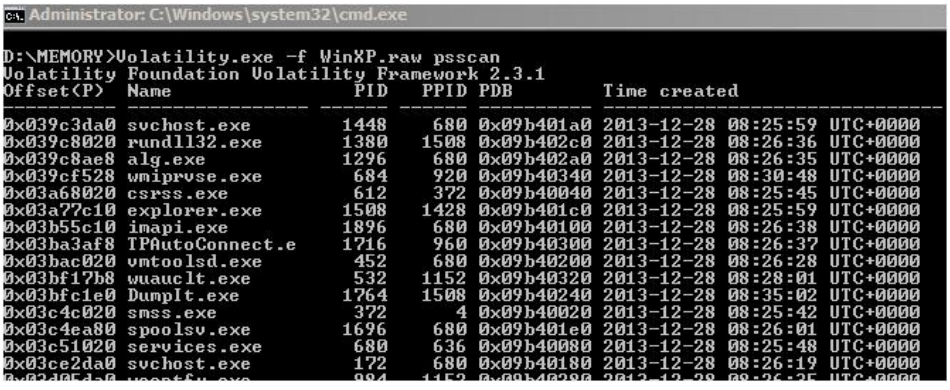


Figure 3. Process Scanning

In above image you can clearly see that after psscan plugin shows all the processes printed out with their corresponding process IDs and Offset. (These offsets and Process IDs are used in malware analysis.

Services Analysis

Services analysis is yet another important analysis for investigator as per forensic prospective. By services analysis, investigator easily knows which services was running last time on the system and which services was not running, even you can clearly see the process IDs of every service. Sometimes investigator detect unwanted services that was running on system which are not related to system or its default services, it may be activated by Trojan, RAT, or by other malicious program.

Figure 4. Scanned Process Snippet

In above image you can see, after service scan (svcscan) we get brief description about services and easily knows that whether they are started or stopped. Analysis of these services are very important because investigator can found malicious activity or evidence from such analysis.

In above image you can see offsets of the processes with correspondence order number and process ID of services as well you can clearly see service name, display name and service type that shows the type of service, means which service runs by user or system, then you can see service state, that shows you service was running or not, if it indicate stop so service was stop in the system. Then you can see path of the service from the service was executes.

Chapter 3 – Registry Basics

Registry is the one of most important part in Windows operating systems. Registry managed in hierarchical form and stores configuration settings of users and systems on Windows operating system. The kernel, device drivers, services, SAM, user interface and third party applications can all make use of the registry. The registry also provides a means to access counters for profiling system performance.

Hives

Registry managed in hierarchical form in Windows to manage settings and data in proper way.
Windows operating system have different 2 hives for managing settings like Machine, Root, Security, User, and Default. Hives are the root directories that stores subdirectories called keys. In other words, integrated hierarchical database, branches of the registry are actually stored in a number of disk files called hives.

Some hives are volatile and are not stored on disk at all. An example of this is the hive of branch starting at HKLM\HARDWARE. This hive records information about system hardware and entry is created each time when the system boots and performs hardware detection.

Not all hives are loaded in one time, when system or application need a hive so that it will load instantly.

*See your hive list by type “regedit” command in run box. Hives name shows in below table and shows their corresponding short form and work. Different hives contains different -2 types of value or settings.

Table. 1. HKEY specification

Name	Abbreviation	Contents
HKEY_CLASSES_ROOT	HKCR	Information used by programs for file association and for sharing information.
HKEY_CURRENT_USER	HKCU	Settings and configuration for the current user.
HKEY_LOCAL_MACHINE	HKLM	Settings and configuration for all users.
HKEY_USERS	HKU	Settings and configuration for all users on the computer; the information in HKCU is copied from this hive when the user logs in.
HKEY_CURRENT_CONFIG	N/A	Hardware information about the PC's resources and configuration.

Volatility Framework provide some plugins to gather information about hives. These plugins work frequently and ease to use.

Hivelist

Hivelist plugin provide all the hives list that are available in registry memory. It shows the virtual addresses or offset of registry hives in memory and the full path of corresponding hive on Windows registry.

Hivedump

Recursively list all subkeys in a hive, use the hivedump command and pass it the virtual address to the desired hive.

Hivescan

If you want to find the all physical address in Windows registry hive. This plugin of volatility framework isn't generally useful by itself because hivelist provide better work done than it.

Keys Information

We already discussed about it in hive section. In other word or take an example of it – Folder A contains Folder B and Folder B contains folder C so in registry case folder A is the hive then folder B is the key and folder C is the sub-key. Let another example: Hives >> Keys >> Subkeys. Means hive contains keys and keys contains subkeys. In below image you can clearly understand what keys are.

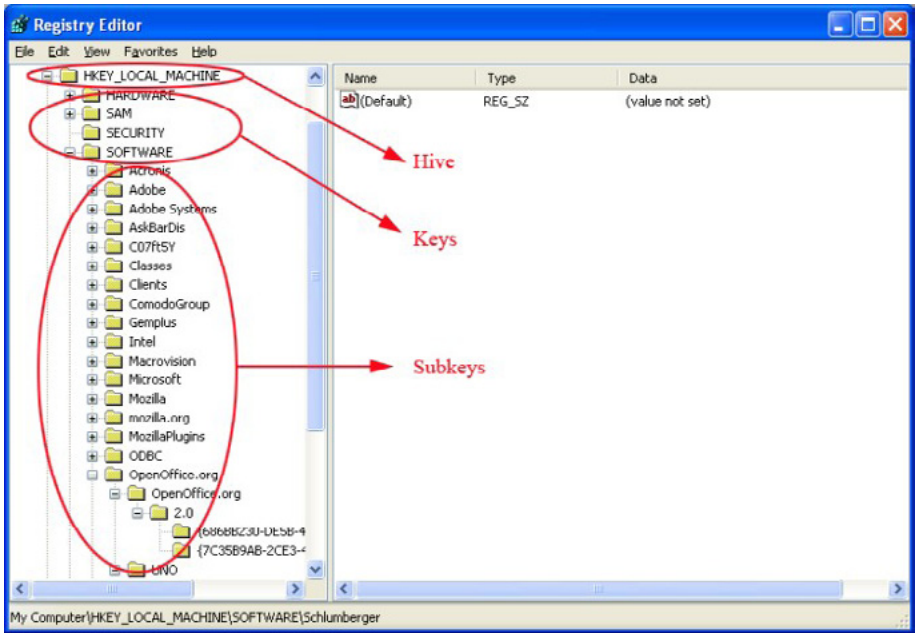


Figure 5. Windows Registry Editor

In above image you can clearly see HKLM hive contains HARDWARE, SAM, SECURITY, SOFTWARE Keys and these key contains subkeys as shown in image.

Navigating Keys and Subkeys in Registry

Volatility Framework provides very useful and easy way to navigate keys and Subkeys in memory by "printkey" plugin. Printkey use to display the subkeys, values, data, and data types contained within a specified registry key. Let's take an example of it, if you want to print all the keys that available in the system hive so:

```
Volatility.exe –f WinXP.raw –profile=WinXPSP2x86 printkey –o hive_offset
```

In place of hive_offset you have to put hive's offset/virtual address of the hive that you can found with the help of *hivelist* plugin or command. After this you will get all the keys that hive contains. Now in more deep, if you want to locate under Key means you want to locate subkeys under keys so for it you have to use –K then key name. For example:

```
Volatility.exe –f WinXP.raw --profile=WinXPSP2x86 printkey –o hive_offset –K Key_Name
```

Some keys and subkeys contains values that stores configuration settings by system, users and network. These values may be in encrypted form so that cannot be easily understandable. Below image shows you an example of locating keys and subkeys in Windows registry.

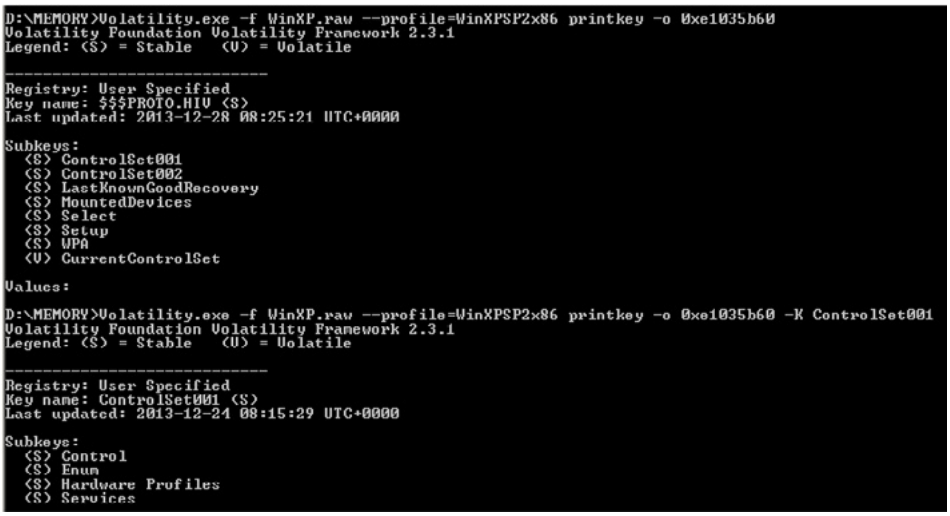


Figure 6. Uses of PrintKey Plugin

Chapter 4 – Hardware Information

Gathering information about hardware from Windows operating system by registry is good idea. Windows OS stores hardware and BIOS information in registry that helps investigator to find information about BIOS and hardware which are connected to the system. In under of HKLM hive, HARDWARE key contains information about hardware and BIOS. I have a dumped image of Windows 7 so I'll use this dumped image to collect information about hardware and BIOS.

CPU Identification

Windows registry is the perfect way to identify central processor unit. In this part we will gather information about CPU. Registry hides value's data in encrypted form and unencrypted form. With the help of hivelist plugin you will get virtual_address/offsets of HARDWARE Key under HKLM as shown below figure.

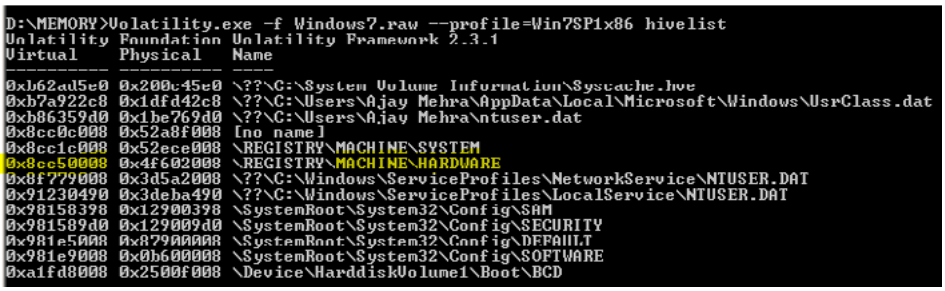


Figure 7. Hives list with hivelist plugins

Copy the virtual address of HKLM/HARDWARE and navigate it with the help of "printkey" plugin –o use for offset/virtual address.

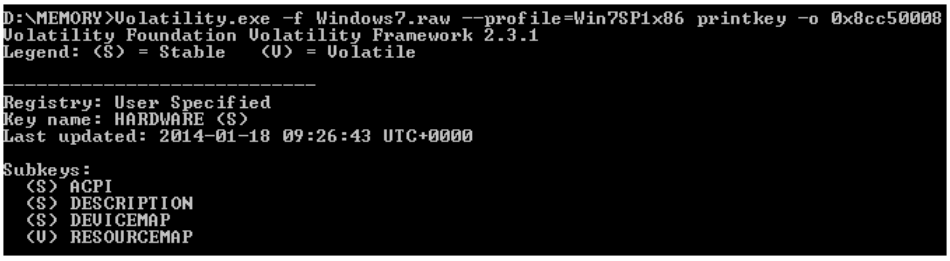


Figure 8. Locating Keys and Subkeys with PrintKey

Now in above image you can see, some subkeys that managed Under HARDWARE Key. Now to gather information about hardware and BIOS so you have to navigate DESCRIPTION/System subkey as shown in below image.

Figure 9. Locating Hives for Hardware Information

As you can see in above image we get little bit information about system by registry. You can see at downside in image SystemBiosDate, SystemBiosVersion, Identifier, BootArchitechture and other information. Now suppose we want to gather more information about CPU so we'll navigate "CentralProcessor". And if we want to gather more information about hardware and BIOS so we have to navigate "BIOS".

First of all we'll navigate CentralProcessor subkey to gather information of CPU and vender. So by using given command you can navigate it:

```
Volatility.exe -f Windows7 --profile=Win7SP1x86 printkey -o HKLM_Virtual -K DESCRIPTION\System\
CentralProcessor\0
```

After executing this command you can clearly see about architecture of OS, processor name, vendor identification and frequency of processor. In my case, processor is Intel Pentium B960 @2.20GHz and in MHz it was 2195 accurately.

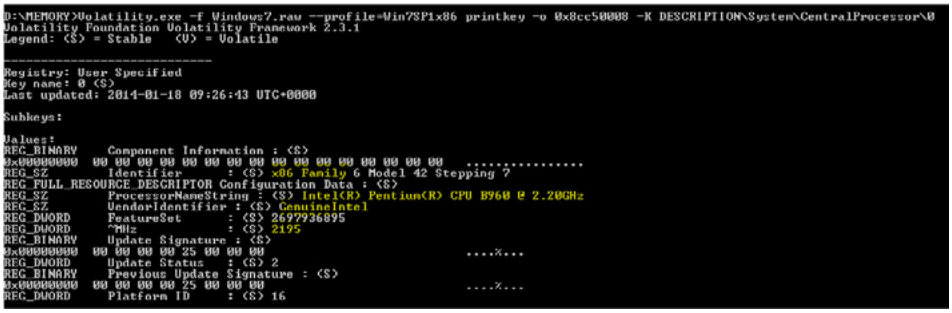


Figure 10. Gather Information about CPU

Hardware and BIOS Identification

In this part we'll look at how to collect information about compromised machine. When some hardware or device driver was detected by Windows registry, automatically Windows create the registry values in HKLM hive. This is good way to collect information about compromised machine.

Let's gather information about compromised machine with the help of dumped memory image. We have to just navigate on the right way with the help of printkey plugin:

```
Volatility.exe -f Windows7.raw -profile=Win7SP1x86 -o HKLM_VirtualAddress -K DESCRIPTION/System/
BIOS
```

After this command you will get important information about hardware and BIOS that uses by compromised system as shown in below image.

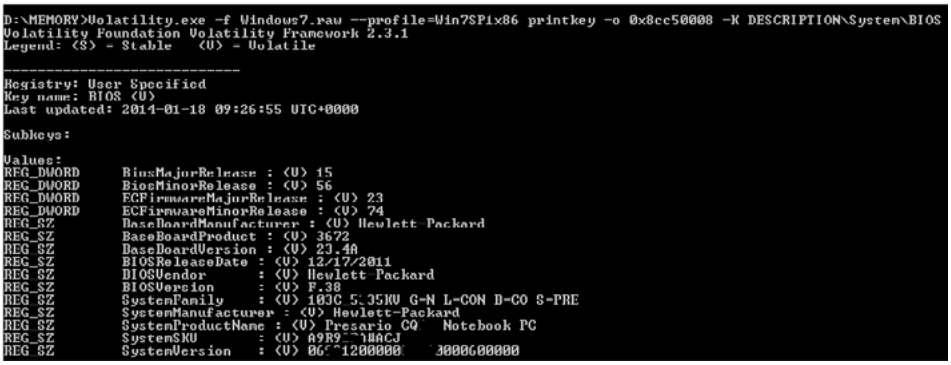


Figure 11. Gather Information about Hardware and BIOS

Note

Small Computer System Interface (SCSI) is a set of standards for physically connecting and transferring data between computers and peripheral devices.

You can get some interesting information from here. Here you can find out BaseBoardManufacturer means notebook manufacturer, BaseBoardProduct number, BaseBoardVersion and now on BIOS, the BIOS release date, BIOS Vendor, BIOS version you can get from here easily. After that System Family, Manufacturer, Product name even you can get serial number, and system version from here.

SCSI Devices

Now if you want to identify or gather information about SCSI devices like Harddisk devices and DVD-Writer and so on, so following command will help you to locate SCSI devices.

```
Volatility.exe -f Windows7.raw -profile=Win7SP1x86 -o HKLM_VirtualAddress -K "DEVICEMAP\Scsi\Scsi Port 0\Scsi Bus 0"
```

Figure 12. Locating SCSI Devices

If any SCSI devices are connected to the system so you can find entry of the SCSI devices here. In above image you can see there are two Target IDs 0 and 1. If there is more Target IDs it means more SCSI devices was connected to the system. If you will navigate these Target IDs then you will get SCSI devices information. As shown below image:

Figure 13. Locate SCSI Devices with their Serial Number

Figure 14. Locating SCSI Devices 2

Chapter 5 – Hash Dumping and LSA Secrets

In this chapter you will learn about hashes and LSA Secrets and some dumping techniques by Volatility Framework.

Hash Dumping

Windows SAM stores password in the unreadable format means in encrypted form or in the form of hashes. Windows user's password stores in the hash form in the SAM database. Windows uses for hash functions LM hash and NTLM hash for securing users password. When a user put their password for login so Windows convert this password in hashes and match with the hash that stores in SAM database. If the hashes are matched then user will be login. And in case the hashes doesn't matched so Windows will give error of incorrect password.

Volatility framework provide a very good and useful plugins to dump all the hashes from SAM database. Investigator can dump all the hashes and decrypt them but sometimes decryption hashes takes time. So for dump hashes what we need??

Volatility Framework provide a plugin namely "hashdump" for dump hashes but we have to fill all the parameters that are necessary for hash dump. For it we need virtual address or offset of SAM hive and System hive that you can get easily with the help of "hivelist" plugin.

In below image you can see i hi-lighted SAM and System with their corresponding virtual addresses or offsets because for hash dumping we have to use them with necessary parameters.

```
D:\MEMORY>Volatility.exe -f WinXPSP2.vmem --profile=WinXPSP2x86 hivelist
Volatility Foundation Volatility Framework 2.3.1
Virtual    Physical   Name
---------- ---------- ----
0xe1c49008 0x036dc008 \Device\HarddiskVolume1\Documents and Settings\LocalService\Local Settings\Application
 Data\Microsoft\Windows\UsrClass.dat
0xe1c41b60 0x04010b60 \Device\HarddiskVolume1\Documents and Settings\LocalService\NTUSER.DAT
0xe1a39638 0x021eb638 \Device\HarddiskVolume1\Documents and Settings\NetworkService\Local Settings\Applicati
on Data\Microsoft\Windows\UsrClass.dat
0xe1a33008 0x01f98008 \Device\HarddiskVolume1\Documents and Settings\NetworkService\NTUSER.DAT
0xe153ab60 0x06b7db60 \Device\HarddiskVolume1\WINDOWS\system32\config\software
0xe1542008 0x06c48008 \Device\HarddiskVolume1\WINDOWS\system32\config\default
0xe1537b60 0x06ae4b60 \SystemRoot\System32\Config\SECURITY
0xe1544008 0x06c4b008 \Device\HarddiskVolume1\WINDOWS\system32\config\SAM
0xe13ae580 0x01bbd580 [no name]
0xe101b008 0x01867008 \Device\HarddiskVolume1\WINDOWS\system32\config\system
0xe1008978 0x01824978 [no name]
0xe1e158c0 0x009728c0 \Device\HarddiskVolume1\Documents and Settings\Administrator\Local Settings\Applicatio
n Data\Microsoft\Windows\UsrClass.dat
0xe1da4008 0x00f6e008 \Device\HarddiskVolume1\Documents and Settings\Administrator\NTUSER.DAT
```

Figure 15. Hi-Lighted SAM and Systems Virtual Address/Offsets

Now we have virtual addresses of SAM and System hives so we will use them for dump hashes.

```
Volatility.exe -f WinXP.raw --profile=WinXPSP2x86 hashdump -y system_virutal -s sam_virtual
```

You have to pass system virtual address with –y and SAM virtual address with –s.

```
D:\MEMORY>Volatility.exe -f WinXPSP2.vmem --profile=WinXPSP2x86 hashdump -y 0xe101b008 -s 0xe1544008
Volatility Foundation Volatility Framework 2.3.1
Administrator:500:e52cac67419a9a224a3b108f3fa6cb6d:8846f7eaee8fb117ad06bdd830b7586c:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HelpAssistant:1000:4e857c004024e53cd538de64deda873b:842b4013c45a3b8fec76ca54e5910581:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:81db385a61425fc7874c3268aa249ea1:::
```

Figure 16. Dumped Hashes from Windows Registry

Now we have dump hashes of all users including administrator.

LSA Secrets

That's a beauty of memory forensics. We can collect a lot of information about compromised system in investigation. Before we start dumping of LSA secret we have to understand about LSA Secrets. So let's start understanding LSA Secrets, may be many of you knows about it but this is a beginners guide so start it from beginning.

LSA Secrets is a special protected storage area for important data used by Local Security Authority (LSA) in Windows. That storage area stores important data like Local Security Policies, Auditing, Authentication, Logging users on the system, Storing private data as well it stores system's and user's sensitive data in secrets. Access of this secret data only available for system. When Microsoft launched Windows, means in starting LSA stores cached domain records but as time goes away Microsoft's developer made a lot implements on LSA Secrets then now LSA Secrets stores Internet Explorer passwords, RAS connection passwords, SQL and CISCO passwords, SYSTEM account passwords, private user data like EFS encryption keys, and a lot more.

Some directories where LSA secrets stored is `HKEY_LOCAL_MACHINE/Security/Policy/Secrets` and the parent directory `HKEY_LOCAL_MACHINE/Security/Policy` contains additional data, necessary for accessing and decrypting the secrets.

Dumping LSA Secrets

Volatility framework's plugin "lsadump" provide great facility to dump LSA Secrets from Windows registry. To dump LSA Secrets we have to pass necessary parameters of Systems and Security's virtual addresses/Offsets.

```
Volatility.exe -f WinXP.raw -profile=WinXPSP2x86 lsadump -y system_offset -s security_offset
```

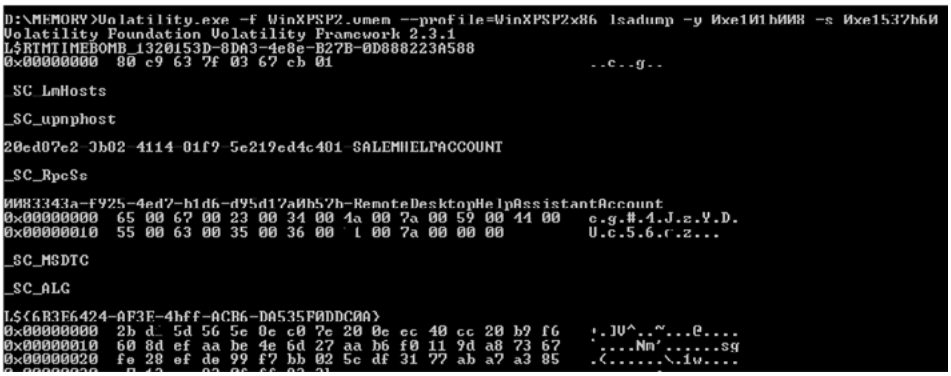


Figure 17. Dumped LSA Secrets from Windows Registry

You can see dumped LSA Secrets in above image. This data is in encrypted form. There first 12 byte contains metadata and remaining data is in encrypted form.

Note

I want to suggest you, if you want more brief information about LSA Secrets so at least go once these articles.

- *http://www.passcape.com/index.php?section=docsys&cmd=details&id=23*
- h*ttp://moyix.blogspot.in/2008/02/decrypting-lsa-secrets.html*

Chapter 6 – Shellbags Analysis

One of the most important part in Windows registry forensics is the Shellbags analysis or forensics. Shellbags analysis is important for Windows registry investigator because investigator can find a lot of information and collect evidence from registry. Shellbags contains most useful data that helps to investigate.

Introduction to shellbags and its analysis

Microsoft Windows uses track users viewing preferences means size, icon, view of folder using Windows explorer, this call Shellbags information and this information stores in Windows registry at different -2 places. Shellbags is all about directories. If somebody open or close a folder so this log/entry also stores in Shellbags even if somebody delete the folder so this entry you can see in registry, which means that investigator can use to enumerate past mounted volumes, deleted files, and user actions.

You can find Shellbags information in registry on these locations:

- HKEY_USERS\{USERID}\Software\Microsoft\Windows\Shell\
- HKEY_USERS\{USERID}\Software\Microsoft\Windows\ShellNoRoam\

Shellbags information stores in registry in encrypted form. I suggest you to look at once on “Using Shellbags Information to re con str uct user activity” research paper.

Now comes on Volatility framework, It is an awesome framework that provide Shellbags analysis by just one plugin “shellbags”. Shellbags plugin extract all the shellbags information into the Windows registry. The output of this plugin is easily readable by user. As shown in below example:

```
Volatility.exe -f WinXP.raw -profile=WinXPSP2x86 shellbags
```

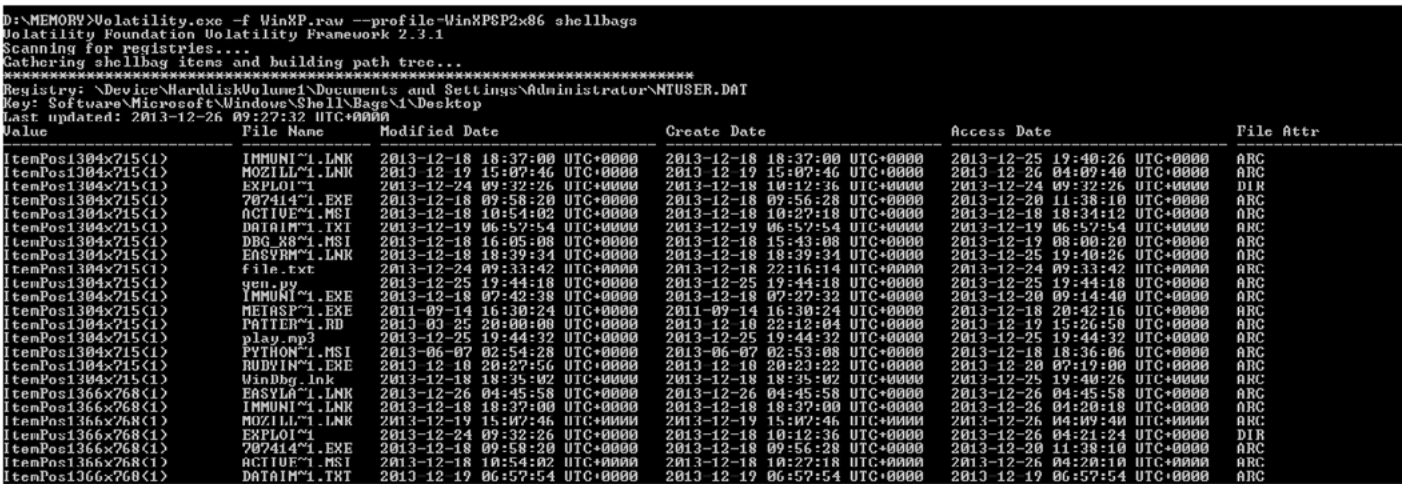


Figure 18. Shellbags Information

In above image you can see a snippet of shellbags plugin. As you can clearly see after this plugin we get a lot of information about directories and the links or file (Links or file because shellbags manage visual preferences like icon, size, view etc.). Now investigator have a lot information about compromised system like when last folder or directory was accessed and when modification, creation done. For example you can see in value section the screen resolution then file name. And in modification, creation and accessed section shows you the activity of directory or file.

I'll suggest you try it atleast once. And if you want to copy all the output by shellbags plugin in text file so use following command:

```
Volatility.exe -f WinXP.raw -profile=WinXPSP2x86 shellbags > shellbags.txt
```

Now everything that you can see in buffer about shellbags information that will store in shellbags text file and you can access it and analyze according to case.

Chapter 7 – UserAssist and Shim Cache Analysis

In this chapter we'll discuss about UserAssist and Shim Cache Analysis. Both analysis are important for Windows Registry Forensics (WRF).

Introduction to UserAssist and Its Analysis

UserAssist Is another important analysis In Windows registry for investigation. Windows operating system stores user activity in registry.

`HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist` key in the Registry. UserAssist shows the program that executed in Windows with last execution (executable and links) time and path of executed program.

Userassist helps the investigator to find out last program execution or which program executed on Windows. These all investigation done according to case. Suppose a malicious .exe file executed on compromised machine so userassist help the investigator to find out evidence.

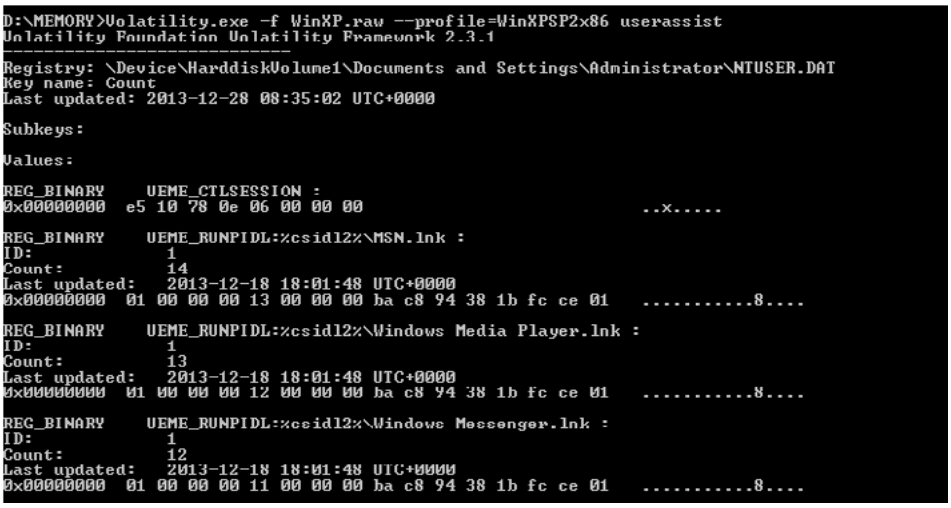


Figure 19. Dumped UserAssist Data for Analysis

```
Volatility.exe -f WinXP.raw -profile=WinXPSP2x86 userassist
```

Volatility framework provide a very good and useful plugin for dump userassist information from Windows registry. By below example you can see we can dump userassist information with the help of “userassist” plugin. In below image you can see some dumped data like REG_BINARY, ID, Count, Last Updated and some encrypted data. The information within the binary, UserAssist values contains only statistical data on the applications launched by the user via Windows Explorer.

Note

If you want to know more about userassist and want to analyze in more better way so I’ll suggest you to read this article. *http://www.aldeid.com/wiki/Windows-userassist-keys*.

Introduction to Shim Cache and Its Analysis

Shim Cache analysis is one of my favorite analysis in Windows registry forensics (WRF). First start from basics, two things you have to understand “what is shim cache?” and second is “how we can analyze it for forensics?” So let’s start from first question what is shim cache?

Shimcache shows all the .exe files that executes in Windows. If a file was executed with Windows to “CreateProcess”, It will logged in Shimcache Key. Also shimcache check application compatibility with Windows explorer. Userassist and Shim Cache mostly helpful malware cases. In other and easy words suppose a person executed Adobe photoshop on Windows OS and this application is perfectly compatible with Windows OS now once when Adobe photoshop will executes on Windows so logs or information will store in Windows registry in shim cache directory.

Volatility framework provide shimcache plugin to dump all the shimcache data. Once when you run this plugin, in the output you will get all the executable that are compatible with Windows and they executed on same Windows machine. These type of all data stores in Windows registry.

```
Volatility.exe -f Win7.raw -f -profile=Win7SP1x86 shimcache
```

```
D:\MEMORY>Volatility.exe -f Win7.raw --profile=Win7SP1x86 shimcache
Volatility Foundation Volatility Framework 2.3.1
Last Modified                 Path
----------------------------- ----
2009-07-14 01:14:22 UTC+0000   \??\C:\Windows\system32\LogonUI.exe
2009-07-14 01:14:35 UTC+0000   \??\C:\Windows\system32\SearchFilterHost.exe
2009-07-14 01:14:35 UTC+0000   \??\C:\Windows\system32\SearchProtocolHost.exe
2011-11-22 21:45:45 UTC+0000   \??\C:\Program Files\WinRAR\WinRAR.exe
2013-06-18 14:21:12 UTC+0000   \??\C:\Program Files\Mozilla Firefox\firefox.exe
2012-08-18 05:38:30 UTC+0000   \??\C:\Program Files\Kaspersky Lab\Kaspersky Anti-Virus 2013\ffcert.exe
2009-07-14 01:16:20 UTC+0000   \??\C:\Windows\System32\wpdshext.dll
2009-07-14 01:14:42 UTC+0000   \??\C:\Windows\system32\taskhost.exe
2012-09-21 06:08:22 UTC+0000   \??\C:\Windows\system32\igfxsrvc.exe
2009-07-14 01:14:42 UTC+0000   \??\C:\Windows\servicing\TrustedInstaller.exe
2009-07-14 01:14:28 UTC+0000   \??\C:\Windows\system32\PING.EXE
2009-07-14 01:14:44 UTC+0000   \??\C:\Windows\system32\w32tm.exe
2009-06-10 21:22:50 UTC+0000   \??\C:\Windows\Microsoft.NET\Framework\v2.0.50727\cvtres.exe
2009-06-10 21:22:49 UTC+0000   \??\C:\Windows\Microsoft.NET\Framework\v2.0.50727\csc.exe
2009-07-14 01:14:47 UTC+0000   \??\C:\Windows\system32\wbem\wmiprvse.exe
2009-07-14 01:14:41 UTC+0000   \??\C:\Windows\System32\svchost.exe
2009-07-14 01:14:43 UTC+0000   \??\C:\Windows\system32\vssvc.exe
2009-07-14 01:14:22 UTC+0000   \??\C:\Windows\system32\lpremove.exe
2009-07-14 01:14:31 UTC+0000   \??\C:\Windows\System32\rundll32.exe
2009-07-14 01:14:35 UTC+0000   \??\C:\Windows\System32\sdiagnhost.exe
2009-07-14 01:15:22 UTC+0000   \??\C:\Windows\System32\gameux.dll
2009-07-14 01:14:28 UTC+0000   \??\C:\Windows\System32\powercfg.exe
```

Figure 20. Dumped ShimCache Information for Analysis

In above image you can see a snippet of shimcache information. Volatility framework's "shimcache" plugin works with Windows 7 Image.

Chapter 8 – Most Recent Used

Now something juicy and interesting information. Most Recent Used (MRU) files. Windows registry stores the information of recently used items in HKCU Hive. Most recent used is one of important forensics methods for forensic investigator. In some cases a file was executed and system got infected or compromised. In this type of cases most recent used files help the investigator to find evidence.

In HKCU hive you can find MRU items in Windows Vista and Windows7.

```
HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\
ComDlg32\CIDSizeMRU
Volatility.exe -f Windows7.raw --profile=Win7SP1x86
printkey -o HKCU_Offset(ntuser.dat) -K
Software\Microsoft\Windows\CurrentVersion\Explorer\Co
mDlg32\CIDSizeMRU
```

```
Legend: (S) = Stable   (V) = Volatile

----------------------------
Registry: User Specified
Key name: CIDSizeMRU (S)
Last updated: 2014-01-24 13:28:18 UTC+0000

Subkeys:

Values:
REG_BINARY   20          : (S)
0x00000000  70 00 79 00 74 00 68 00 6f 00 6e 00 77 00 2e 00  p.y.t.h.o.n.w...
0x00000010  65 00 78 00 65 00 00 00 00 00 00 00 00 00 00 00  e.x.e...........
0x00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0x00000030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0x00000040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 21. Most Recent Used Dumped

In above image shows a snippet of MRU. You can see some encrypted data with recent used file. Some data is clearly understandable like in this image it shows most recent visited file was pythonw.exe.

Try these all stuff by yourself.

References

- Wikipedia.org
- Volatility Framework
- DumpIt
- My Mind

About the Author

Kapil Soni is a youngest security analyst and researcher. He is enthusiastic about cyber and technology world. He works with many organizations and works as a penetration tester and trainer. Recently he is a Co-founder of Expert Tech Community (Non-Profit Open-Source Community based on knowledge sharing) and Chief Editor of ToolWar.

Kali Linux Primer

by Chad Oliver

Kali Linux is the latest version of the BackTrack Linux penetration testing, security auditing, and forensics distribution. It is based on Debian and comes ready to go with all the tools you need to begin an information security engagement.

The amount of tools available in the distribution prevents us from going into depth on each tool, but this tutorial is designed to get you started with some of the most common tools you will use to perform a typical security audit.

For the purposes of this tutorial we will be running a known vulnerable OS called Metasploitable which is available at *http://www.offensivesecurity.com/metasploitunleashed/Metasploitable* and we are focusing on network penetration testing. Kali however has much more to offer, including application testing via tools such as Burpsuite, and SQL Injection tools such as sqlmap. For Social Engineering engagements it is complete with tools such as Maltego for doing some excellent reconnaissance, BeEF for attacking browsers (think XSS), and it includes the Social Engineering Toolkit.

Getting Started

Many of the tools are command line based and clicking a tool in the menu will open a terminal window and show you the the help for using the tool.

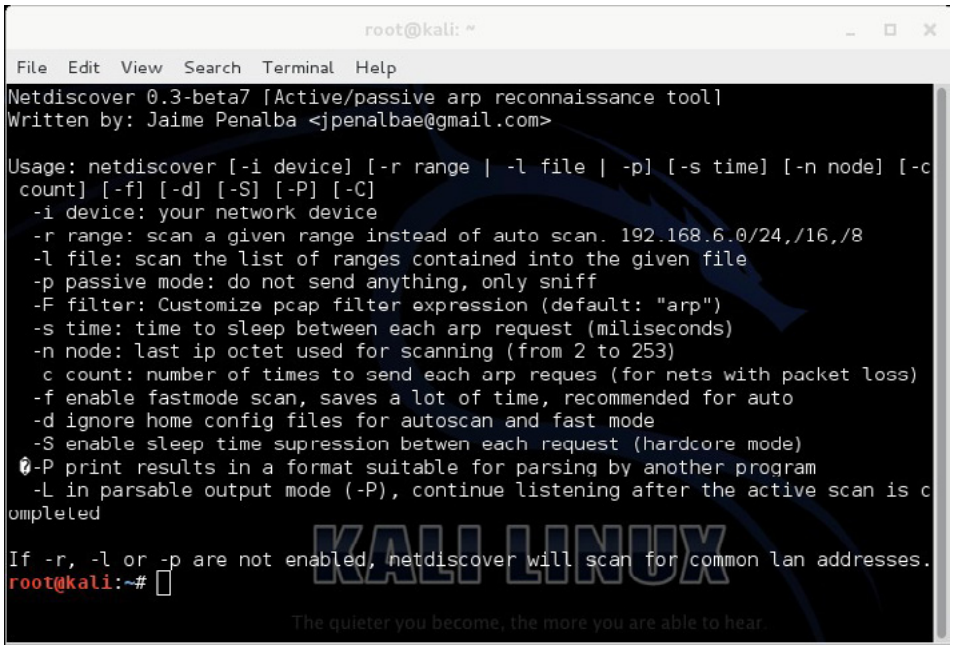


Figure 1. Terminal window

Let's begin by using netdiscover to see what is is on the network without being intrusive. We can run it in passive mode so it only sniffs the traffic it sees and doesn't send anything out. Depending on the rules of engagement, you may want to try to stay hidden during your test and this will get you started identifying machines on the network.

```
Currently scanning: (passive)   |   Screen View: Unique Hosts

85 Captured ARP Req/Rep packets, from 5 hosts.   Total size: 5100

  IP            At MAC Address      Count  Len   MAC Vendor
-------------------------------------------------------------------------
192.168.1.1     20:4e:7f:8d:64:d6    53    3180   Unknown vendor
192.168.1.119   08:00:27:9c:4b:37    06    360   CADMUS COMPUTER SYSTEMS
192.168.1.128   00:1f:5b:3e:a7:c3    08    480   Apple, Inc.
192.168.1.140   54:26:96:4d:fe:7f    15    900   Unknown vendor
192.168.1.116   b8:8d:12:0b:7e:74    03    180   Unknown vendor
```

Figure 2. Using as Unknown Vendor

The second entry is our metasploitable instance. This doesn't give us much to go on but you can see a couple devices on the network and their manufacturer if it's known. So let's take a look at the traffic and see what we find.

Watching the Traffic

Wireshark is a fantastic tool for watching, capturing, and analyzing network traffic. I've personally used it on very large incident responses to data breaches and used it to track down conversations between exploited systems and the external attackers. It quickly let's you identify the protocol being used and the IP addresses of the machines communicating. Once you've identified something worth investigating you can right click on the capture and set up filters to watch specific conversations, IP addresses, protocols, etc.

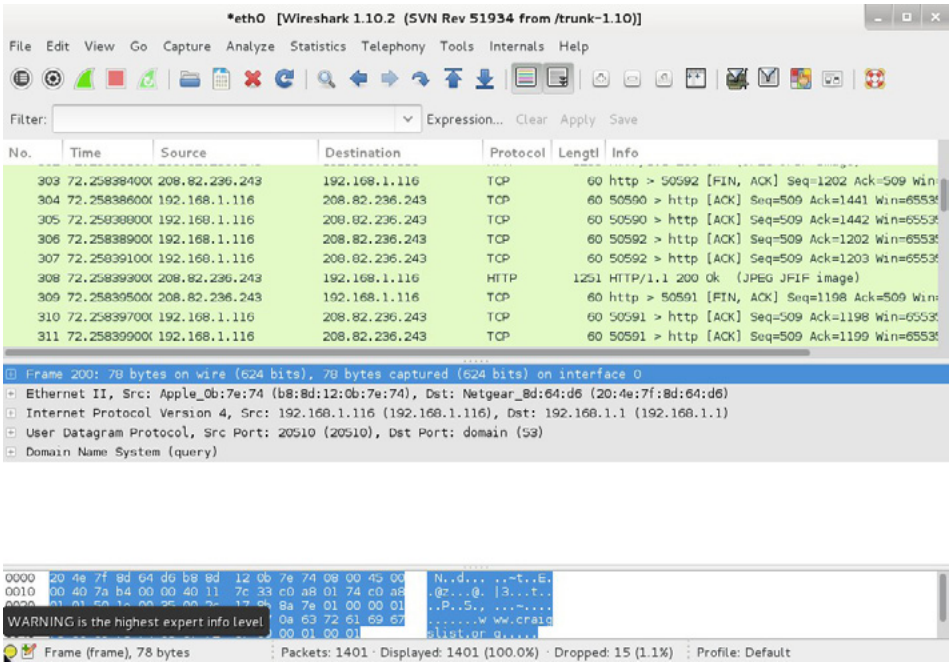


Figure 3. Investigating

Wireshark not only shows you the basic information about the traffic, it has two additional sections below that let you analyze each packet and see exactly what is being transmitted. When performing a penetration test it is useful to let it run for a long time and then go back through the data examining protocols used, active IP addresses, etc. Any protocols that can be used to transmit credentials in clear text are a great score such as telnet and http. For example, if you find a network router that has the web interface enabled, it's certainly worth a look to see if you can catch someone logging in. Depending on your rules of engagement, it may be a little dirty pool, but you could set Wireshark capturing traffic while you call help desk and report some connectivity issues to try to get them to log into the router. Remember that any traffic is worth examining regardless of how mundane it may appear. During an incident response with a live APT in the

network I noticed some peculiar DNS traffic and discovered the attackers had been using DNS to exfiltrate data by having the compromised machines look up things such as sensitivecustomerdata.evilattackershost.cc. It wasn't quite that simple, but that gives you an idea.

Fingerprinting Devices

Before we start making noise on the network, let's talk about a tool that can fingerprint passively, p0f. It's a great tool in general to run to watch your network just to monitor health. It will tell you what devices are live, their OS, what type of connection it has, distance, and uptime. I like to run it in promiscuous mode so executing p0f p gets us identifying systems without making a sound. In our test environment I have found the Metasploitable system in the results.

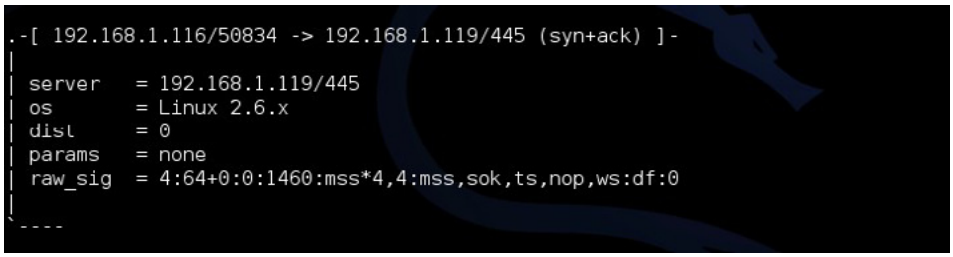


Figure 4. Foundings

We now know a little bit more about this particular server, for example it's running Linux 2.6.x. We can start identifying machines in this way to help us zero in our target. For example, if your client is running some old outdated OS (like Windows 2000 for example) this may help you find a vulnerable system without ever having made a noise on the network. So far, we have only been operating in promiscuous mode and not sending any packets out on the network that might trip an IDS. As with all the tools mentioned in this article and all the other tools available on Kali Linux, you can do a little research to find lots of additional features that will help you do your job.

Let's take a moment and start making a little noise. Of course it is possible to zero in a target without making a wave but for the most part (unless you are intentionally evading the IDS as part of the test) your client is more interested in learning what you can discover in the shortest amount of time to get the biggest bang for their buck. It's time we break out nmap. Nmap is a network mapping tool that is going to cut right through all the mess and tell us just what is running on our network, what OS, what ports are open, what services and versions are running, etc. It is feature rich and I encourage you to visit nmap.org and learn everything there is about this amazing contribution to security world. There is a wealth of options to run nmap including slowing the scan down to a crawl to help evade detection. Doing so will take a very long time. You can also increase the speed to get the task down as fast as possible. Depending on your engagement, check the options available and run what works best for you. You may find yourself running several scans with various options to help speed along your productivity.

Taking nmap a step further and giving us the benefit of GUI, there is zenmap. Zenmap let's us quickly use nmap and shows us the results in an easier to digest format. Here we have a sample of the available ports on our metasploitable instance.

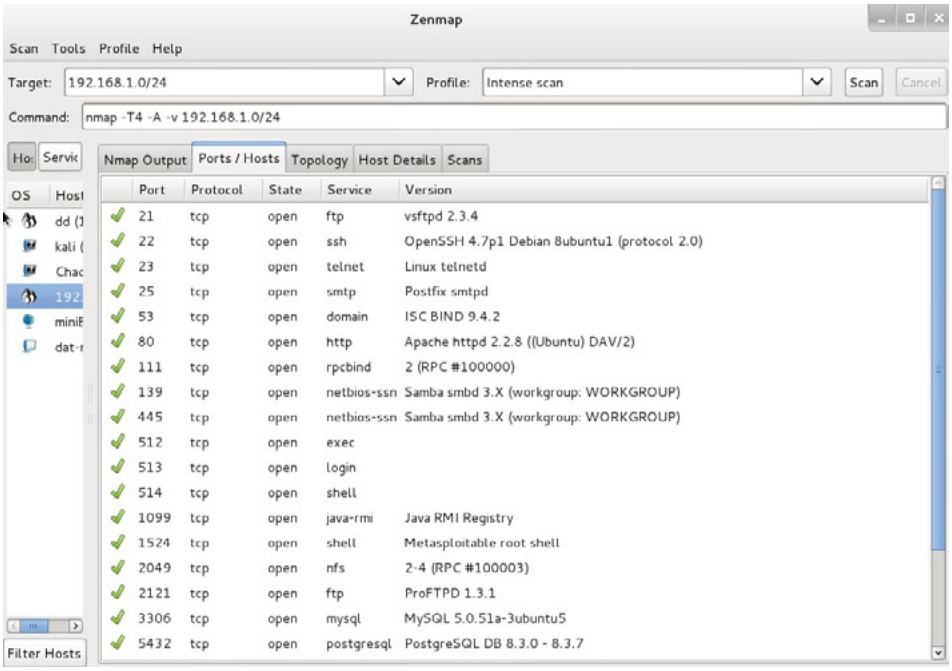


Figure 5. Available ports

In addition, the Topology feature of Zenmap gives os a quick glance of some insightful data. The legend is available at *http://nmap.org/book/zenmaptopology.html* but generally speaking it's using a a green, yellow, red, color scheme to identify number of open ports. Red being most. If a host is identified as a router or switch it is indicated by square, otherwise it is a circle. We can see our Metasploitable system as a red circle. On a large network this can help you quickly identify a system that may be misconfigured with everything open, or at least we can assume it is something with a lot of roles to fill and we may find some good services ripe for exploiting. Of course, in the case of our example this is obviously true.

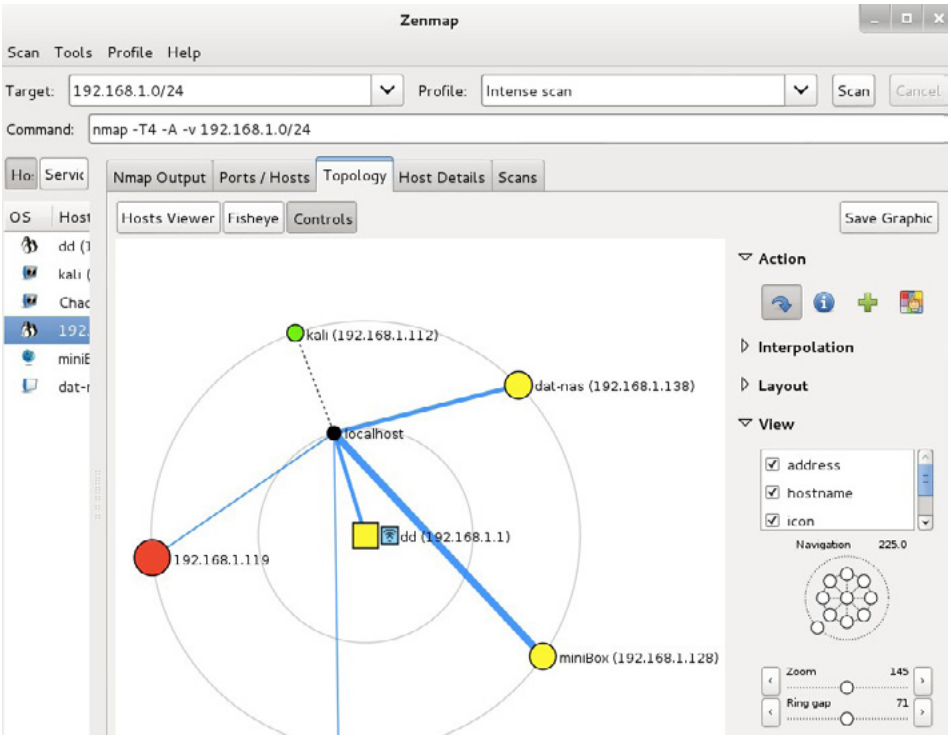


Figure 6. Example of network

Nmap/Zenmap gives us a good view of our target and we could do some online research on our services version information to find what vulnerabilities exist and if there are exploits available. Kali Linux even helps us out here by providing a tool to search for exploits, searchsploit.

Using our port information for our Metasploitable instance, we can run through each and see if we find anything. For example, running *searchsploit bind 9.4.2* gives us a hit.

```
root@kali:~# searchsploit bind 9.4.2
 Description                                                              Pat
h
---------------------------------------------------------------------------- ----
---------------------
BIND 9.4.1-9.4.2 - Remote DNS Cache Poisoning Flaw Exploit (meta)          /mul
tiple/remote/6122.rb
root@kali:~#
```

Figure 7. Running searchploit bind 9.4.2.

If our engagement involves a vulnerability assessment Kali Linux provides us with a great tool for conducting assessments, OpenVAS. OpenVAS allows you to setup target lists, create schedules, define what sort of tests to be run, and of course execute the test. I've run OpenVAS against our Metasploitable instance to see what it finds. To do so, we first setup our target in the Targets tab at the bottom right of the GUI. Next we switch to the Tasks tab and set a task of running the scan against the target. For the purposes of the demo, I set the target to the Metasploitable target and the set the Scan Config option to "Full and very deep ultimate." Depending on your engagement you can adjust the Scan Config. You can even switch to the Scan Configs tab and define your own scan configs to do things like only run certain types of checks. This may be helpful for doing scans based on compliance guidelines. While we're exploring some of the options, it's worth mentioning there is a tab to supply credentials to the scan which will enhance your results. If your client wants a deeper more accurate report they may supply you with domain/root credentials so the scanner can get onto the system and get better results. It may also be the case that while you were watching traffic in Wireshark you managed to get some credentials which you can now enter into the scanner to get more info on that device and quite possibly, if the credentials are used elsewhere you can get into those systems too. Once our scan is complete, OpenVAS presents us with a nice dashboard showing some highlights of the scan results.

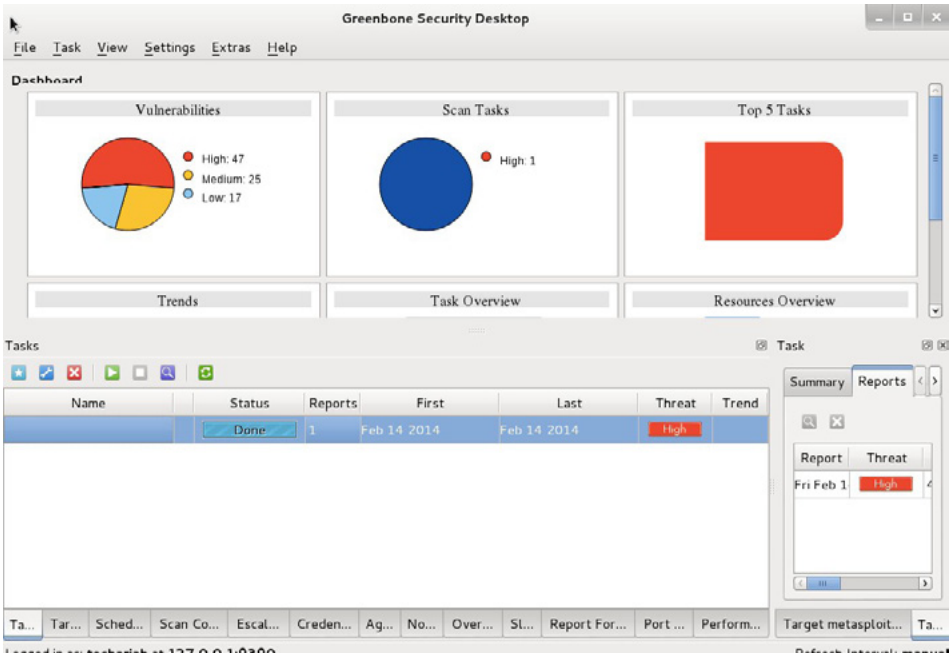


Figure 8. OpenVAS dashboard

We can quickly see there are 47 High Risk vulnerabilities found during our scan. Clicking on the Reports tab in the bottom right of the screen and then the magnifying glass button we can see our results. open in a new tab. From here, we can view the results or we can choose various export formats and save the report.

In general, I usually export as PDF to include in my findings document I turn into the client. The reports are not flashy, but they get the data you need.

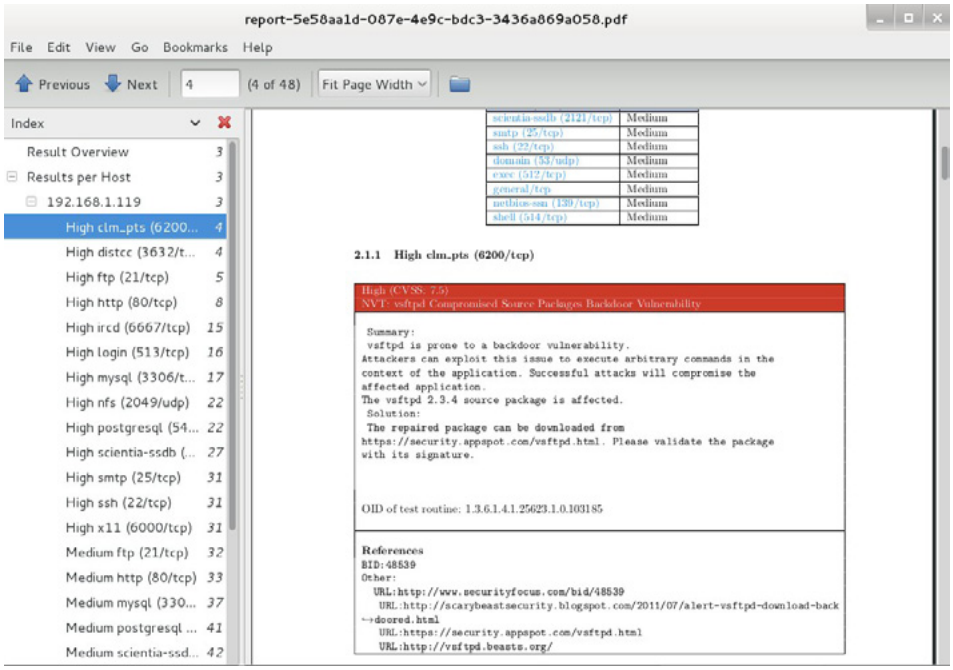


Figure 9. Datas

The layout is pretty self explanatory and the Index included in the PDF can help you navigate right to the vulnerability you want to investigate further. The screenshot shows the first on in the list. In this case we have a backdoor vulnerability in vsftpd. Included in all results is a summary if the issue, a solution to resolve it (if available), and some reference links to learn more about the vulnerability which can be used to find an actual exploit for the vulnerability. Now that we have a vulnerability, let's go ahead and take a look at an exploit.

Exploiting Targets

Previously, we found a bunch of vulnerabilities on our Metasploitable instance. No surprise considering it's designed that way. Let's take a look at the above mention issue and see how we can exploit it. In this case, we can do it without any additional tools. There is a simple backdoor that is ready to exploit. Simply putting an emoticon smiley face at the end of the user name opens a listening shell on port 6200. Let's try.

```
root@kali:~# telnet 192.168.1.119 21
Trying 192.168.1.119...
Connected to 192.168.1.119.
Escape character is '^]'.
220 (vsFTPd 2.3.4)
user oops:)
331 Please specify the password.
pass doesnotmatter
^]

telnet> quit
Connection closed.
root@kali:~# telnet 192.168.1.119 6200
Trying 192.168.1.119...
Connected to 192.168.1.119.
Escape character is '^]'.
id;
uid=0(root) gid=0(root)
```

Figure 10. Port 6200

We open a terminal and telnet to port 21. Once connected we simply enter *user oops:)* followed by *pass* with anything you choose as it doesn't matter. Escape the session and telnet to port 6200 and we are in.

That wasn't too terribly exciting, but mission accomplished and had we been operating with an effort to remain stealthy we could have discovered that and exploited it all with limited ability to detect our attack. Let's take a look at an exploit tool. Metasploit. Metasploit is an exploit framework designed for penetration testing. It has several interfaces that use the command line, and the latest version has a web interface. There is also a commercial version, but that is obviously not going to come ready to go with Kali Linux. Armitage is included in Kali Linux and gives us a nice interface for an attack platform.

It uses nmap and other tools to help us find targets. Start Armitage and it will connect to metasploit if you have it running, otherwise it will start the service for you. Additionally, it will provide you with some help when starting such as telling you how to start the database if that is not already running. Once we get it open, we're ready to begin. Since we are focusing on this single server let's go ahead and add the host to Armitage. Under the Hosts menu we can click Add Host and enter the IP address of our target.

Once our host is added to Armitage it is represented by a monitor in the main workspace. We can right click on it and select scan to get some more data about it. Doing this opens a new tab on the bottom and we can watch the scan in action. Once complete, we can see what the OS is and the services running on it. Similar to our findings using nmap. Right clicking the server and choosing Services opens us a new tab and lists the available services. You can see the services below and it has given us a visual reference in the workspace identifying the OS of the host. This can help you when looking at a large range of hosts to identify various operating systems you may want to dive into.

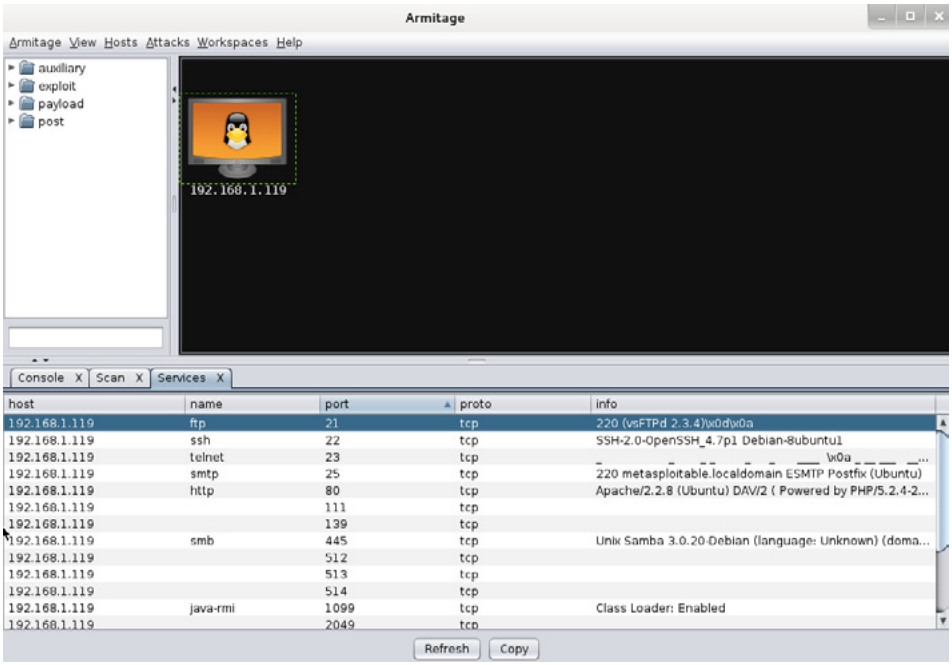


Figure 11. Hosts

From here, we can search the upper left area for an exploit to use against the services. For example, going back to the previous exploit which is highlighted in the above screenshot, I have searched for vsftp and found our exploit is available in Metasploit.

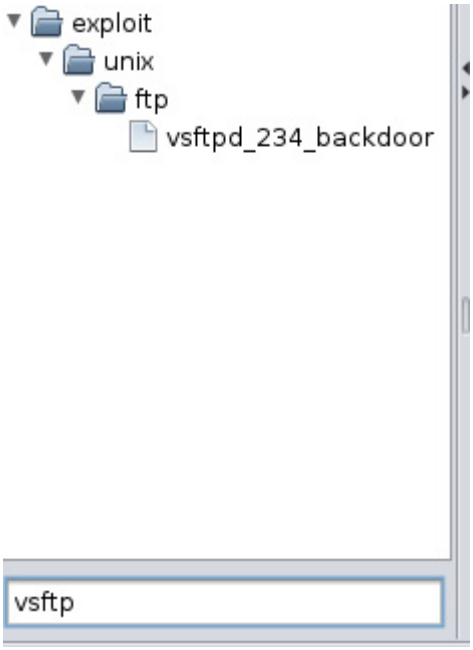


Figure 12. Exploits

Double clicking our exploit brings up the ability to launch the exploit. Note that launching an exploit this way does not fill in all the data needed to execute. In this case, we need to supply the IP of our target in RHOST. Once we do that and click the launch button a new tab is opened at the bottom and it shows us our exploit in action. We can see the exploit was successful and notice the icon in the workspace has changed to represent an exploited system.

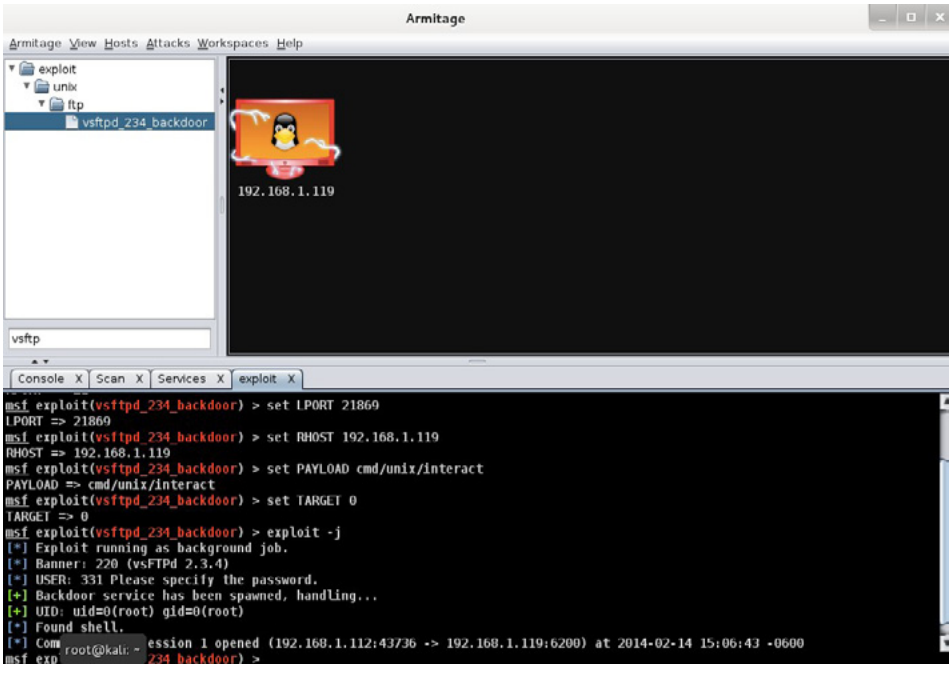


Figure 13. Icon in the workspace

Right clicking the host in the workspace shows us a new option in the menu. This is is called Shell 1 and going to that we can select Interact which will gives a new tab at the bottom showing us our shell. I have run ifconfig to verify the IP is our target system.

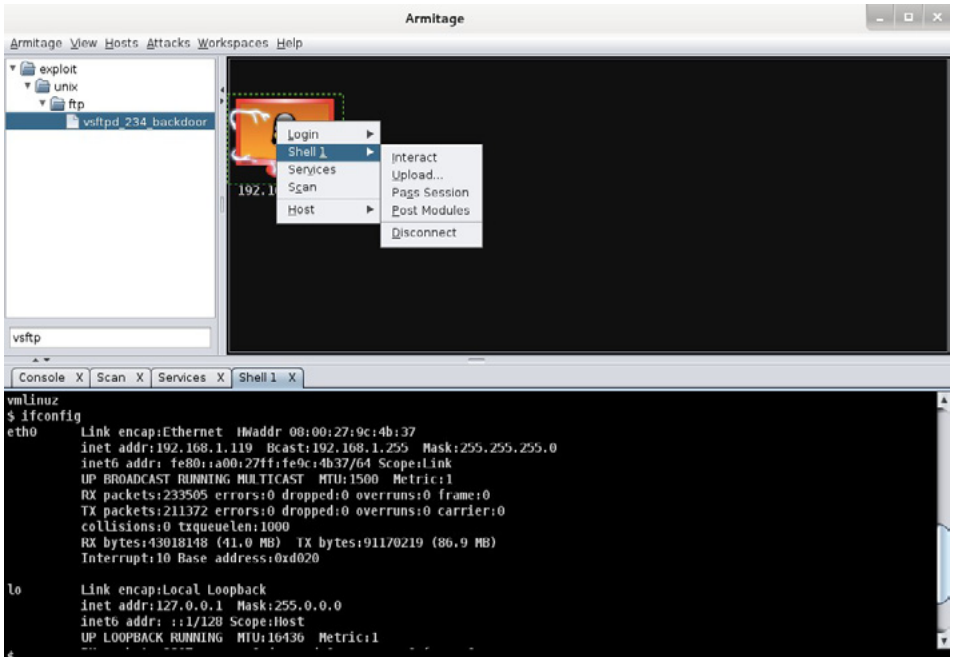


Figure 14. Selecting Interact

WE can take advantage of another feature of armitage to help us find exploits. Under the Attack menu there is the option to Find Attacks. Doing this will automate the process of finding an attack suitable for our target and give us a new option in the pop up menu when right clicking our target in the workspace. using this Attack menu, we can launch an attack directly with the attack data prefilled and ready to go.

While we are looking at the main Attack menu, you may have noticed the option called Hail Mary located under the Find Attacks option. This option will launch a flood of attacks trying everything it can to infiltrate the selected targets. Using this option will light up an IDS system so if you haven't already, make sure you are whitelisted on their IDS system so you don't overwhelm the system. After launching the Hail Mary option it will open a new tab below and show all the attacks being launched. When it's complete it will list the shells it opened and right clicking our target in the workspace shows us a list of shells we can then select to interact with.

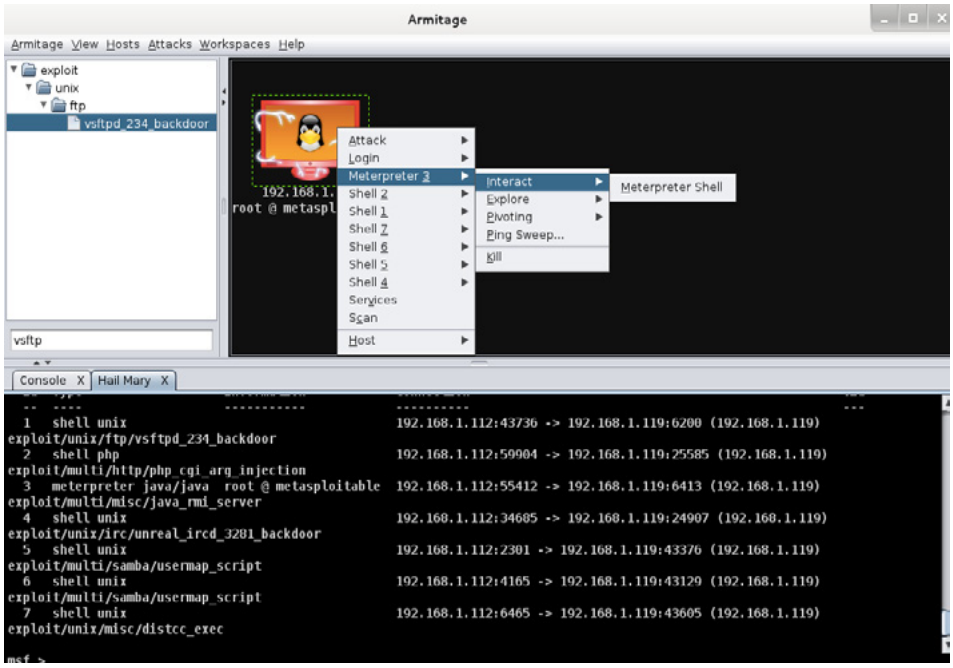


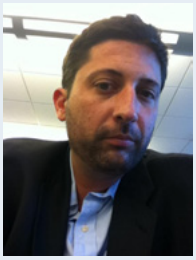
Figure 15. List of shells

From this point, we can use the compromised machine to pivot and launch attacks against other systems on the network through this compromised host, all while remaining inside our Armitage work environment.

Conclusion

Kali Linux is a versatile tool and should be a part of every penetration testers arsenal. It is a complete attack tool and can be run as a Live CD (which is excellent if you are working on a forensics project), and be installed directly to a machine or virtual machine. Of course, you can always add any other tools you come across that are not part of Kali. I have been in this business for years, and BackTrack/Kali is my OS of choice on all my penetration testing engagements. We've only scratched the surface of what it has to offer and I hope you've enjoyed this introduction and I encourage you to dive in and learn every tool it has to offer.

About the Author



Chad Olivier is the owner of Shades of Gray Security. He has over fifteen years of experience in IT security. He has worked in every industry performing social engineering, reverse engineering, penetration testing, vulnerability research, and incident response. He has never met a client he could not social engineer. www.linkedin.com/in/chadolivier | www.shadesofgraysecurity.com

3RD REGIONAL IT INFRASTRUCTURE OPERATIONS & GREEN DATA CENTER CONFERENCE 2014

"A Sustainable Approach towards Innovation, Infrastructure and Operations"

April 23rd & 24th, 2014 | Kuala Lumpur, Malaysia

"We shall require a substantially new manner of thinking if mankind is to survive."

-Albert Einstein

Guest of Honour: ***Y.B Datuk Seri Panglima Dr. Maximus Johnity Ongkili****, Minister, Ministry of Energy, Green Technology and Water (KeTTHA)*

CONFERENCE FOCUS

This 3rd Regional IT Infrastructure Operations & Green Data Center Conference is to provide you with ***"Sustainable and efficient operations are critical to the success of businesses today."*** especially on:

- **Global pressures** to reduce carbon emissions
- **Reducing** data center operation & management cost
- **Revolutionising** energy efficiency
- **Develop** Server Density and Power Consumption
- **Managing** Explosive Growth
- **Perfecting** Green IT technology
- **Latest initiative:** Cloud computing
- **Creative solutions** in finding qualified staff – urgent need to reduce outsourcing
- **Stability** of smart energy systems.
- **Optimization** of energy-efficient protocols.

WHO SHOULD ATTEND

Top level decision makers such as CEOs, COOs, CTOs, CIOs, Heads, President, VPs, Directors, GMs and senior personnel including Local Government, Agencies & Authorities of:

- Head of IT
- Head of IT Infrastructure
- Head of Information Security
- Data Centre Manager & Engineer
- Data Centre Facilities Manager & Engineer
- Data Centre Operations Manager & Engineer
- Disaster Recovery Manager
- Business Continuity Manager
- Network, Cabling and Communication Engineers and Integrators
- Server, Storage and Application Administrators

From various industries

SBL CLAIMABLE

Organised by:

COMFORI

Registered with:

PSMB APPROVED TRAINING PROVIDER CLASS A (Serial No: 1631)

MINISTRY OF FINANCE (MOF No: 357-02054304)

CELEBRATING 10 year CULTIVATING YOUR EMPLOYEES' POTENTIAL

BEST TRAINING PROVIDER Award

SME Best Awards

100% Money Back Guarantee

Dr.WEB®
since 1992

Dr.Web 9.0

for Windows — the rapid response anti-virus

1. Reliable protection against the threats of tomorrow
2. Reliable protection against data loss
3. Secure communication, data transfer and Internet search

www.drweb.com

Free 30-day trial: https://download.drweb.com

New features in Dr.Web 9.0 for Windows: http://products.drweb.com/9

FREE bonus — Dr.Web Mobile Security:
https://download.drweb.com/android

UPDATE NOW WITH STIG AUDITING

"IN SOME CASES **nipper studio** HAS VIRTUALLY **REMOVED** the NEED FOR a **MANUAL AUDIT**"

CISCO SYSTEMS INC.

Titania's award winning Nipper Studio configuration auditing tool is helping security consultants and end-user organizations worldwide improve their network security. Its reports are more detailed than those typically produced by scanners, enabling you to maintain a higher level of vulnerability analysis in the intervals between penetration tests.

Now used in over 45 countries, Nipper Studio provides a thorough, fast & cost effective way to securely audit over 100 different types of network device. The NSA, FBI, DoD & U.S. Treasury already use it, so why not try it for free at www.titania.com

2012 Computing Security Awards WINNER Enterprise Security Solution of the Year

Security Products' GOVIES 2013 Government Security Awards

2012 Computing Security Awards WINNER Network Security Solution of the Year

www.titania.com